REMARKS

In the Office Action, the Examiner rejected the claims under 35 USC §103. These rejections are fully traversed below. In addition, the claims have been amended to correct minor informalities and to further clarify the subject matter regarded as the invention. Claims 1-54 remain pending.

Reconsideration of the application is respectfully requested based on the following remarks.

CLAIM REJECTIONS UNDER 35 U.S.C. §103(a)

Claims 1-42 were rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,333,269 (Calvignac) in view of U.S. Patent No. 5,920,568 (Kurita). In addition, claims 43-54 are rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent No. 5,333,269 (Calvignac) in view of U.S. Patent No. 5,920,568 (Kurita) and further in view of U.S. Patent No. 5,099,517 (Gupta).

As described in Applicant's specification, the traditional router has several potentially undesirable characteristics. First, an inbound port of a router typically has a single inbound queue associated therewith. Second, when a packet is forwarded, a single packet or entry in the inbound queue is transferred by an inbound controller to an outbound controller. Third, when the packet is received by the outbound controller, information associated with a single packet is stored in an outbound queue.

Each of the independent claims provides at least one of the following advantages or limitations over the prior art. First, a plurality of inbound queues is provided for a single inbound port. An inbound packet is therefore classified in one of the plurality of inbound queues to enable the inbound packet to be stored in the appropriate queue. Second, one of the plurality of inbound queues is transferred to an outbound controller and/or outbound queue capable of

storing (or identifying) a multiplicity of inbound queues. In other words, a queue of packets rather than a single packet is transferred to the outbound controller and/or an associated outbound queue. Third, in some embodiments of the invention, it is possible to encrypt an inbound queue prior to transmission by an outbound controller. One or more of the above-described limitations are present in each of the claims.

For instance, independent claim 1 recites a method for providing an inbound controller for a router having an inbound port and an outbound port. More specifically, claim 1 recites, in relevant part, providing a plurality of inbound queues for an inbound port...classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria, storing the inbound packet in the selected one of the plurality of inbound queues, and determining when one of the plurality of inbound queues is ready to be moved to an outbound queue capable of storing a multiplicity of inbound queues.

Calvignac relates to a mechanism for transferring messages between source and destination users through a shared memory. See Title. More specifically, the mechanism is a device that includes a common bus to which a memory with a plurality of independent buffers, a memory interface and a central control apparatus are connected. The memory interface receives messages from source users, stores the messages in selected buffers and chains the buffers together. The central control apparatus generates inbound message queues and outbound message queues in response to commands which it receives from the memory interface. See Abstract.

Calvignac neither discloses nor suggests a mechanism for use with a router having at least one inbound port and at least one outbound port. Moreover, Calvignac neither discloses nor suggests a method for providing or implementing an inbound or outbound controller for a router. More particularly, Calvignac neither discloses nor suggests an outbound queue that is capable of storing or otherwise identifying a plurality of inbound queues. Rather, as the Examiner indicates, Calvignac discloses a prior art outbound queue that stores a plurality of messages, rather than storing or otherwise identifying a plurality of inbound queues. It is important to note that a queue is a data structure that is used to store a plurality of packets. Similarly, as the Examiner admits, Calvignac neither discloses nor suggests transferring one of the plurality of inbound queues (storing a plurality of packets) to such an outbound queue, and therefore neither discloses nor suggests determining when one of a plurality of inbound queues is ready to be moved to an outbound queue. Rather, col. 12, lines 26-27 indicate that Calvignac merely discloses enqueueing a single message onto an outbound queue, and therefore teaches away from transferring or enqueueing an entire queue of packets. In addition, Calvignac neither discloses

nor suggests providing a plurality of inbound queues for an inbound port and classifying a packet in one of the plurality of inbound queues.

It is important to note that the apparatus disclosed in Calvignac does not have a plurality of inbound ports and outbound ports. Specifically, Figure 8 of Calvignac illustrates a data movement receive circuit (86-RCV). See, e.g., col. 17, lines 48-50. In addition, Calvignac illustrates a data movement transmit circuit (86-XMIT). See, e.g., col. 23, lines 15-16. This device appears to include only a single RCV and XMIT. Thus, even if the operation of these circuits were similar to that of an inbound and/or outbound port, the apparatus illustrated only includes a single such circuit for processing incoming and outgoing messages, respectively.

Calvignac discloses link inbound queues. See, e.g., col. 2, lines 30-34. However, the link inbound queues are not associated with an inbound port (or outbound port). Rather, the inbound queues appear to be associated with a source user. Specifically, col. 27, lines 16-18 state that "the messages received from one source user 12 are chained and enqueued into the user link inbound queue."

Calvignac does disclose the handling of messages stored in an inbound queue. However, Calvignac neither discloses nor suggests transferring a queue of packets to an entry in an outbound queue. Rather, Calvignac states that "microprocessor may process the messages...and then enqueue them on the link outbound queue by sending an enqueue global order." See, e.g., col. 12, lines 32-47. Similarly, col. 2, lines 38-52 of Calvignac state:

> "[w]hen a link inbound queue becomes not empty, the memory interface 22 sends a dequeue order request to the centralized control means, said request identifying the corresponding user queue control block. The message address is provided in response thereto with the identification of the queue control block of the destination user. Then, the memory interface 22 sends and enqueue request to the centralized control means, said request **identifying the address of the message to be enqueued** and the queue control block of the destination user. The processing of this enqueue request by the centralized control means causes the messages to be enqueued in an outbound queue from which it is transferred to the destination user, by memory interface 22. (Emphasis added)"

Thus, a single message from the inbound queue is enqueued, rather than a queue storing a plurality of packets.

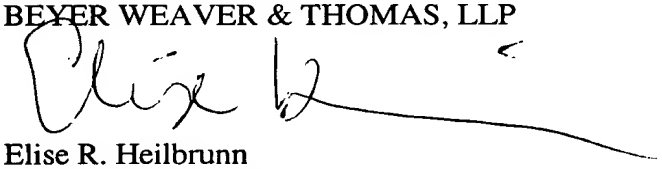
Gupta and Kurita fail to cure the deficiencies of the primary reference. Specifically, Kurita merely discloses a scheduling apparatus and scheduling method. Thus, Kurita neither discloses nor suggests one of the claimed methods of providing an inbound or outbound controller for a router. Moreover, Kurita fails to disclose the transferring of an entire queue to an outbound queue. On the contrary, col. 9, lines 43-45 indicate that a single packet is "picked up" and transmitted. Therefore, Kurita also teaches away from the presently claimed invention.

None of the references, separately or in combination, disclose or suggest the problems present in the prior art or the solutions presented by the presently claimed invention. Based on the foregoing, it is submitted that the independent claims are patentable over the cited references. In addition, it is submitted that the dependent claims are also patentable for at least the same reasons. For instance, dependent claim 2 recites asserting an interrupt when it is determined that one of the plurality of inbound queues is ready to be moved to an outbound queue. Similarly, claim 45, which depends from claim 2, recites transferring the one of the plurality of inbound queues to the outbound queue or an outbound controller associated with the outbound queue when the interrupt is asserted. In other words, an inbound queue may be transferred to an outbound queue or an outbound controller by a CPU, as recited in claim 46. The additional limitations recited in the independent claims or the dependent claims are not further discussed as the above discussed limitations are clearly sufficient to distinguish the claimed invention from the cited references. Thus, it is respectfully requested that the Examiner withdraw the rejection of the claims under 35 USC §103(a).

In view of the above, Applicants believe that all pending claims are allowable and respectfully request a Notice of Allowance for this application from the Examiner. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

If any fees are due in connection with the filing of this amendment, the Commissioner is authorized to charge such fees to Deposit Account 50-0388 (Order No. CISCP054). A duplicate copy of the transmittal is enclosed for this purpose.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP



Elise R. Heilbrunn
Registration No. 42,649

BEYER WEAVER & THOMAS, LLP
P.O. Box 778
Berkeley, CA 94704-0778
Tel: (510) 843-6200

APPENDIX – Marked up copy of claims

Please **AMEND** the claims as follows:

1. (Twice Amended) A method for providing an inbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the inbound controller being adapted for receiving an inbound packet at the inbound port, the method comprising:

providing a plurality of inbound queues for the inbound port;

receiving an inbound packet at the inbound port;

classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

storing the inbound packet in the selected one of the plurality of inbound queues; and

determining when one of the plurality of inbound queues storing a plurality of packets is ready to be moved to an entry in an outbound queue associated with the outbound port, the outbound queue being capable of storing a multiplicity of inbound queues such that each of the multiplicity of inbound queues is stored in one of a plurality of entries in the outbound queue, each of the multiplicity of inbound queues storing a plurality of packets.

2. The method as recited in claim 1, further including:

asserting an interrupt when it is determined that one of the plurality of inbound queues is ready to be moved to an outbound queue.

3. The method as recited in claim 1, wherein classifying the inbound packet includes:

selecting inbound packet sorting criteria;

obtaining packet sorting data for the inbound packet, the packet sorting data being associated with the packet sorting criteria; and

sorting the inbound packet into one of the plurality of inbound queues according to the packet sorting data.

4. The method as recited in claim 1, the selected one of the plurality of inbound queues corresponding to one of a plurality of outbound queues.

5. The method as recited in claim 1, wherein storing the inbound packet includes:

obtaining an available packet buffer from a free pool of available packet buffers;

placing the inbound packet in the packet buffer; and

storing the packet buffer in the inbound queue.

6. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

determining whether a number of packets in one of the plurality of inbound queues exceeds a maximum number of packets.

7. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

determining whether a number of bytes in one of the plurality of inbound queues exceeds a maximum number of bytes.

8. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue further includes:

determining whether a free pool of available memory has been depleted.

9. The method as recited in claim 1, wherein determining when one of the plurality of inbound queues is ready to be moved to an outbound queue further includes:

determining whether a maximum time limit has been exceeded.

10. (Twice Amended) A method for providing an outbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the outbound controller being adapted for forwarding packets at the outbound port, the method comprising:

providing an outbound queue associated with the outbound port and being capable of storing a plurality of inbound queues;

receiving a notification to handle an inbound queue, the inbound queue storing a plurality of packets; and

transferring the inbound queue storing a plurality of packets to the outbound queue associated with the outbound port such that the inbound queue storing a plurality of packets is stored in one of a plurality of entries in the outbound queue.

11. The method as recited in claim 10, wherein receiving the notification includes:

receiving a notification from the CPU to handle the inbound queue.

12. The method as recited in claim 10, further including:

transmitting packets stored in the outbound queue.

13. The method as recited in claim 10, wherein transmitting packets includes:

selectively discarding packets stored in the outbound queue.

14. The method as recited in claim 10, wherein transmitting packets stored in the outbound queue further includes:

obtaining a next one of the plurality of inbound queues stored in the outbound queue;

transmitting selected packets stored in the next one of the plurality of inbound queues; and

releasing memory associated with the next one of the plurality of inbound queues.

15. The method as recited in claim 14, wherein releasing the memory includes:

storing the released memory in a free pool of available packet buffers.

16. The method as recited in claim 14, wherein releasing the memory includes:

forming a new inbound queue to be used by an inbound controller.

17. The method as recited in claim 14, wherein releasing the memory includes:

forming a queue to be used by the outbound controller during bi-directional operation.

18. The method as recited in claim 10, wherein transferring the inbound queue to the outbound queue further includes:

ascertaining a priority of the inbound queue; and

transferring the inbound queue to the outbound queue according to the priority of the inbound queue.

19. (Once Amended) A method for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

providing a plurality of inbound queues for one of the plurality of inbound ports;

providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of storing a plurality of inbound queues;

receiving an inbound packet at the one of the plurality of inbound ports;

classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

storing the inbound packet in the selected one of the plurality of inbound queues;

repeating the steps of receiving, providing, classifying, and storing until an interrupt is asserted; and

transferring one of the plurality of inbound queues storing a plurality of packets to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such that the one of the plurality of inbound queues storing a plurality of packets is stored in one of a plurality of entries in the one of the plurality of outbound queues.

20. (Twice Amended) An inbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the inbound controller being adapted for receiving an inbound packet at the inbound port, comprising:

a packet receiving module coupled to the inbound port, the packet receiving module being adapted for receiving an inbound packet;

wherein the memory has stored therein:

a plurality of inbound queues for the inbound port;

a classifier adapted for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

a packet storing module coupled to the classifier, the packet storing module being adapted for storing the inbound packet in the selected one of the plurality of inbound queues; and

a module adapted for determining when one of the plurality of inbound queues is ready to be moved to an entry in an outbound queue associated with the outbound port, the outbound queue being capable of storing a multiplicity of inbound queues, each of the multiplicity of inbound queues being stored in one of a plurality of entries in the outbound queue and storing a plurality of packets.

21. The inbound controller as recited in claim 20, further including:

a module adapted for providing the determined one of the plurality of inbound queues.

22. The inbound controller as recited in claim 20, further including:

a module adapted for asserting an interrupt when it is determined that one of the plurality of inbound queues is ready to be moved by the CPU to the outbound queue.

23. The inbound controller as recited in claim 20, wherein the packet storing module includes:

a memory obtaining module adapted for obtaining an available packet buffer from a free pool of available packet buffers;

a module adapted for placing the inbound packet in the packet buffer; and

a module adapted for storing the packet buffer in the inbound queue.

24. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a number of packets in one of the plurality of inbound queues exceeds a maximum number of packets.

25. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a number of bytes in one of the plurality of inbound queues exceeds a maximum number of bytes.

26. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a free pool of available memory has been depleted.

27. The inbound controller as recited in claim 20, wherein the module adapted for determining when one of the plurality of inbound queues is ready to be moved to an outbound queue includes:

a module adapted for determining whether a maximum time limit has been exceeded.

28. (Twice Amended) An outbound controller for a router, the router having an inbound port and an outbound port, a memory, and a CPU, the outbound controller being adapted for forwarding packets at the outbound port, comprising:

a module adapted for receiving a notification to handle an inbound queue associated with the inbound port, the inbound queue storing a plurality of packets;

wherein at least one of the CPU and the memory are adapted for storing an outbound queue associated with the outbound port, the outbound queue being capable of storing a plurality of inbound queues in a plurality of entries in the outbound queue, each of the plurality of inbound queues storing a plurality of packets; and

a queue transferring module adapted for transferring the inbound queue storing a plurality of packets to an entry in the outbound queue.

29. The outbound controller as recited in claim 28, wherein the module adapted for receiving the notification includes a module adapted for receiving the notification from the CPU.

30. The outbound controller as recited in claim 28, further including:

a module adapted for transmitting packets stored in the outbound queue.

31. The outbound controller as recited in claim 30, wherein the module adapted for transmitting packets includes:

a module adapted for selectively discarding packets stored in the outbound queue.

32. The outbound controller as recited in claim 30, wherein the module adapted for transmitting packets stored in the outbound queue includes:

a module adapted for obtaining a next one of the plurality of inbound queues stored in the outbound queue;

a packet transmission module adapted for transmitting selected packets stored in the next one of the plurality of inbound queues; and

a memory releasing module adapted for releasing memory associated with the next one of the plurality of inbound queues.

33. The outbound controller as recited in claim 32, wherein the memory releasing module includes:

a module adapted for storing the released memory in a free pool of available packet buffers.

34. The outbound controller as recited in claim 32, wherein the released memory forms a new inbound queue to be used by an inbound controller.

35. The outbound controller as recited in claim 32, wherein the released memory forms a queue to be used by the outbound controller during bi-directional operation.

36. The outbound controller as recited in claim 28, wherein the queue transferring module is adapted for transferring the inbound queue to the outbound queue according to a priority of the inbound queue.

37. (Once Amended) A router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, comprising:

an inbound controller coupled to one of the plurality of inbound ports, the inbound controller being adapted for receiving an inbound packet;

wherein the memory has stored therein:

a plurality of inbound queues for the one of the plurality of inbound ports, each one of the plurality of inbound queues being capable of storing a plurality of packets; [and]

a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of storing a plurality of inbound queues in one of a plurality of entries in the one of the plurality of outbound queues; and

a classifier coupled to the inbound controller, the classifier being adapted for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet

sorting criteria, the selected one of the plurality of inbound queues being associated with one of the plurality of outbound queues;

wherein the inbound controller is adapted for storing the inbound packet in the selected one of the plurality of inbound queues.

38. The router as recited in claim 37, further including:

an outbound controller coupled to the inbound controller;

wherein the inbound controller selects one of the plurality of inbound queues to be transferred to the outbound controller;

wherein the outbound controller is adapted for storing the selected one of the plurality of inbound queues in one of the plurality of outbound queues associated with the packet sorting criteria and transmitting packets stored in the one of the plurality of outbound queues.

39. The router as recited in claim 37, wherein the inbound controller further includes:

a memory obtaining module coupled to the classifier, the memory obtaining module being adapted for obtaining memory for an inbound packet to permit the inbound packet to be stored in the selected one of the plurality of inbound queues in which the inbound packet is classified.

40. The router as recited in claim 38, wherein the outbound controller further includes:

a memory releasing module adapted for releasing selected packet buffers associated with packets stored in the one of the plurality of outbound queues.

41. The router as recited in claim 40, wherein the memory further includes a free pool of available packet buffers and the memory releasing module is adapted for releasing the selected packet buffers into the free pool.

42. The router as recited in claim 38, wherein the outbound controller further includes:

a memory releasing module adapted for providing a new inbound queue to the inbound controller to replace the selected one of the plurality of inbound queues.

43. (Twice Amended) An encryption system, comprising:

an inbound controller adapted for receiving an inbound packet;

a classifier coupled to the inbound controller and adapted for classifying and storing the inbound packet in one of a plurality of inbound queues;

an outbound controller adapted for receiving the one of the plurality of inbound queues, the one of the plurality of inbound queues storing a plurality of packets [queue]; and

an encryption box coupled to the outbound controller, the encryption box being adapted for encrypting the one of the plurality of inbound queues [queue] to provide an encrypted inbound queue to the outbound controller for transmission.

44. (Once Amended) The encryption system as recited in claim 43, wherein the outbound controller includes an outbound classifier adapted for classifying the encrypted inbound queue in one of a plurality of outbound queues associated with a plurality of outbound ports, the outbound controller adapted for transmitting data stored in the one of the plurality of outbound queues.

45. The method as recited in claim 2, further comprising:

when the interrupt is asserted, transferring the one of the plurality of inbound queues to the outbound queue or an outbound controller associated with the outbound queue.

46. The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to the outbound queue or an outbound controller associated with the outbound queue is performed by the CPU.

47. The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to the outbound queue or an outbound controller associated with the outbound queue comprises:

transferring the one of the plurality of inbound queues to an outbound queue corresponding to a priority associated with the one of the plurality of inbound queues.

48. The method as recited in claim 45, wherein transferring the one of the plurality of inbound queues to the outbound queue or an outbound controller associated with the outbound queue comprises:

transferring the one of the plurality of inbound queues to an outbound queue associated with the one of the plurality of inbound queues.

49. The method as recited in claim 43, wherein the inbound queue stores therein a plurality of packets, and wherein the encryption box does not encrypt each of the plurality of packets.

50. The method as recited in claim 43, wherein the encryption box is adapted for encrypting the inbound queue as an entity such that a single encryption step is performed.

51. The method as recited in claim 10, wherein transferring the inbound queue to the outbound queue is performed by the CPU in response to an interrupt.

52. The method as recited in claim 1, wherein the outbound queue comprises a plurality of entries, each of the plurality of entries storing or identifying one of the multiplicity of inbound queues.

53. (Once Amended) A computer-readable medium storing thereon computer-readable instructions for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

instructions for providing a plurality of inbound queues for one of the plurality of inbound ports;

instructions for providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of storing a plurality of inbound queues such that each of the plurality of inbound queues is stored in one of a plurality of entries in the one of the plurality of outbound queues;

instructions for receiving an inbound packet at the one of the plurality of inbound ports;

instructions for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

instructions for storing the inbound packet in the selected one of the plurality of inbound queues;

instructions for repeating the steps of receiving, providing, classifying, and storing until an interrupt is asserted; and

instructions for transferring one of the plurality of inbound queues to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such that the transferred one of the plurality of inbound queues is stored in one of a plurality of entries in the one of the plurality of outbound queues.

54. (Once Amended) An apparatus for forwarding a packet in a router, the router having a plurality of inbound ports and a plurality of outbound ports, a memory, and a CPU, the method comprising:

means for providing a plurality of inbound queues for one of the plurality of inbound ports;

means for providing a plurality of outbound queues, each one of the plurality of outbound queues corresponding to one of the plurality of outbound ports and being capable of storing a plurality of inbound queues such that each of the plurality of inbound queues is stored in one of a plurality of entries in the one of the plurality of outbound queues;

means for receiving an inbound packet at the one of the plurality of inbound ports;

means for classifying the inbound packet in a selected one of the plurality of inbound queues according to packet sorting criteria;

means for storing the inbound packet in the selected one of the plurality of inbound queues;

means for repeating the steps of receiving, providing, classifying, and storing until an interrupt is asserted; and

means for transferring one of the plurality of inbound queues to one of the plurality of outbound queues corresponding to the packet sorting criteria when the interrupt is asserted such that the transferred one of the plurality of inbound queues is stored in one of a plurality of entries in the one of the plurality of outbound queues.